Lightning Talks

Jim Weichel - Session Chair

Presented by Students from ISSGC'09

Open Science Grid

Lightning Talks

- **Malina Kim** - Tier-3s and CMS analysis
- **Andrew Younge** - Towards Efficiency Enhancements in Cloud Computing
- **Anwar Mamat** - Real-Time Divisible Load Scheduling for Cluster Computing
- **Derek Weitzel** - Porting Bioinformatics to the OSG
- **Cole Brand** - Experiences and Difficulties Implementing a Cluster in an Unprepared Environment


Open Science Grid

Tier-3s and CMS analysis

Malina Kirn

University of Maryland

'Base' Analysis* CMS Tier-3

Tier-3 capability	Needed services/software
Service interactive jobs	Interactive node(s), CMSSW
Service local batch jobs	Worker nodes, CMSSW, scheduler
Submit grid jobs	Interactive node(s), CMSSW, CRAB
Store official data	PhEDEx

- Data processed with local batch system.
- Considered a Tier-3 because has access to official data.

*Not all USCMS Tier-3s are analysis oriented. Many devote significant resources to storage or to producing official data.

Open Science Grid

‘CE’ analysis CMS Tier-3

Additional Tier-3 capability	Needed services/software
Service private/unofficial grid production jobs	CMSSW, worker nodes, CE

- All of the services of a base Tier-3 + a CE
- Increasingly rare
- Official data still processed with local batch system (can’t be accessed by grid utilities without an SE).

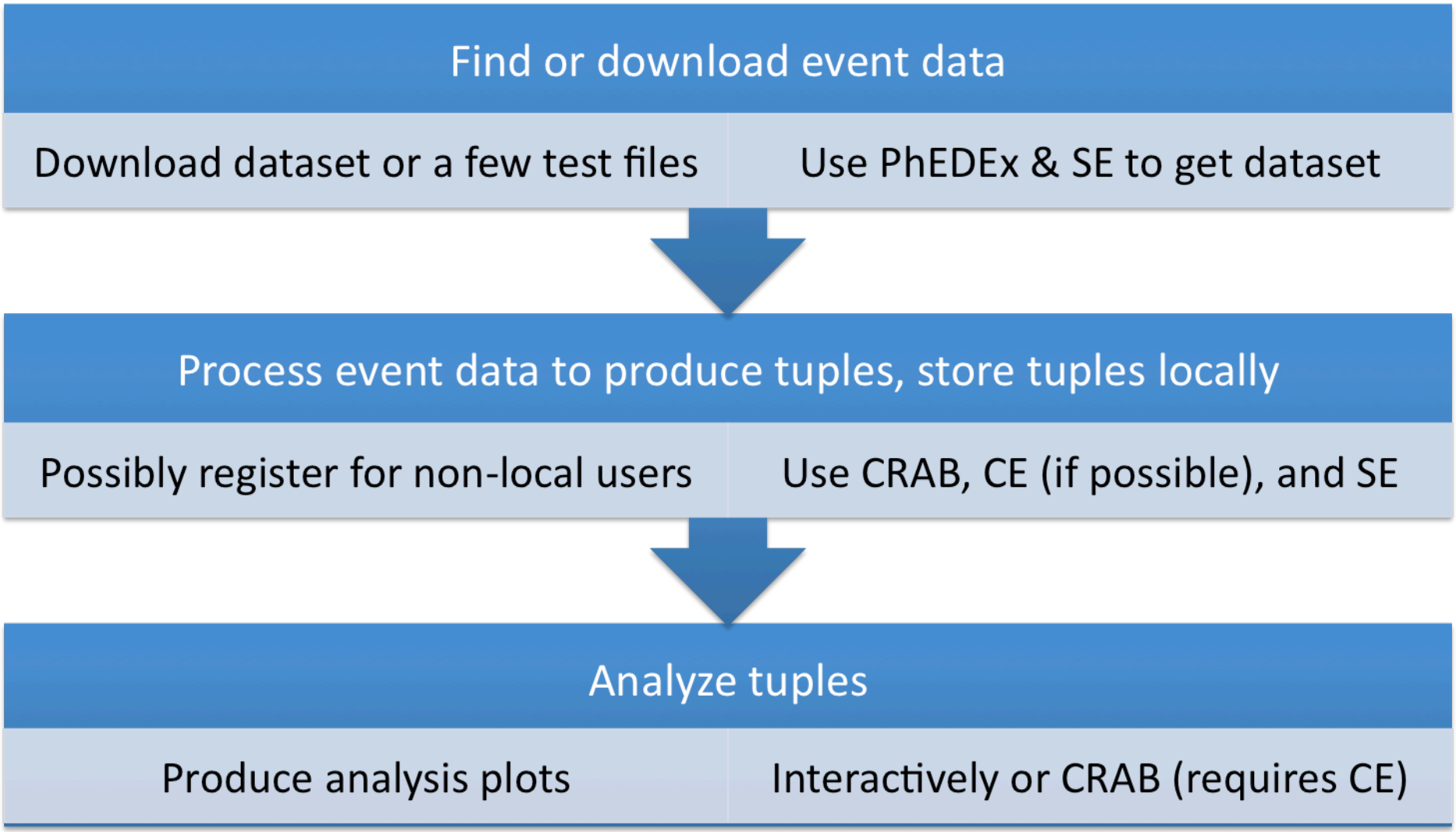
Open Science Grid

‘Fully featured’ Analysis CMS Tier-3

Additional Tier-3 capability	Needed services/software
Store registered private/unofficial data	SE
Service grid analysis jobs	CMSSW, worker nodes, CE, SE

- All of the services of base & CE Tier-3s + an SE
- Users don’t have to learn local batch system (use CRAB).
- Affiliated users, many located at FNAL or CERN, don’t have to interactively login to utilize resources.
- Data stored locally can be utilized by non-local users. Especially convenient for private/unofficial data, which cannot be transferred from the host site (easily).

Analysis process used by students at the University of Maryland




Open Science Grid

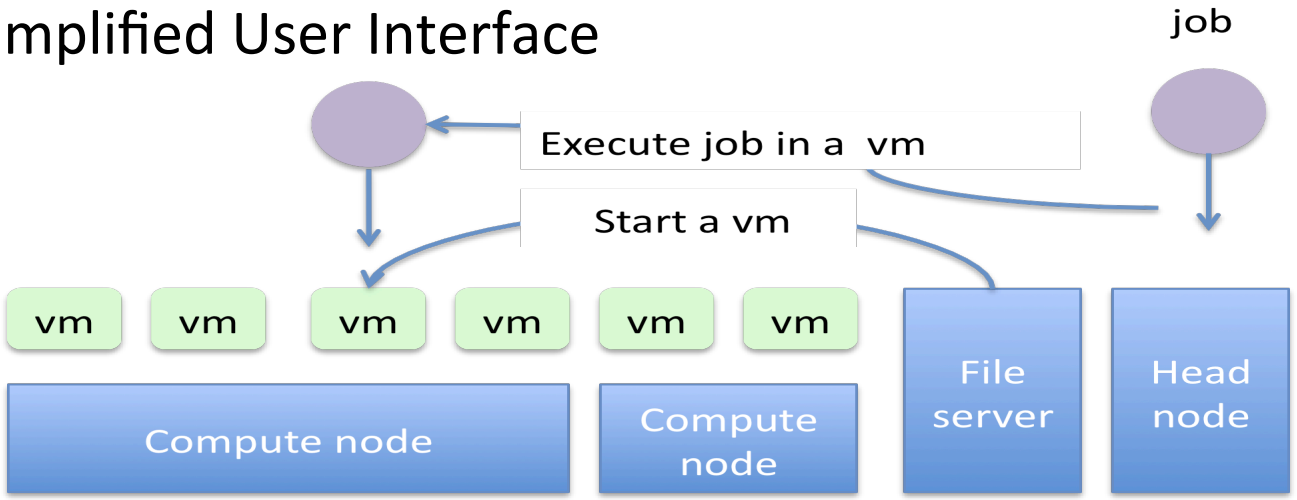
Towards Efficiency Enhancements in Cloud Computing

Andrew J. Younge
Rochester Institute of Technology

http://ajyounge.com

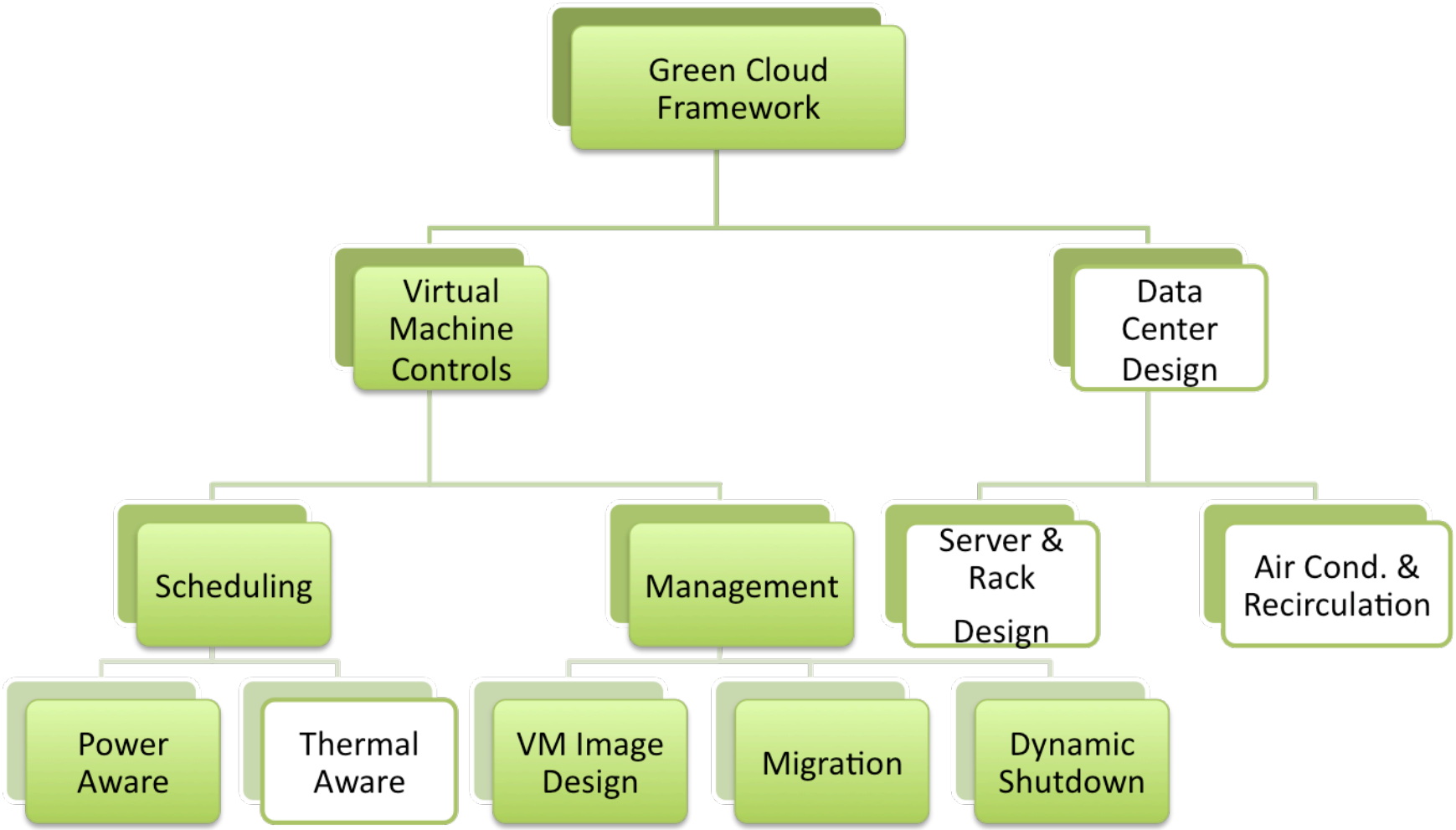
Cloud Computing

- Features of Clouds
  - Scalable
  - Enhanced Quality of Service (QoS)
  - Specialized and Customized
  - Cost Effective
  - Simplified User Interface
- Scientific Cloud computing has become a reality.
- Provides customized frameworks and services to users at little additional cost.



Open Science Grid

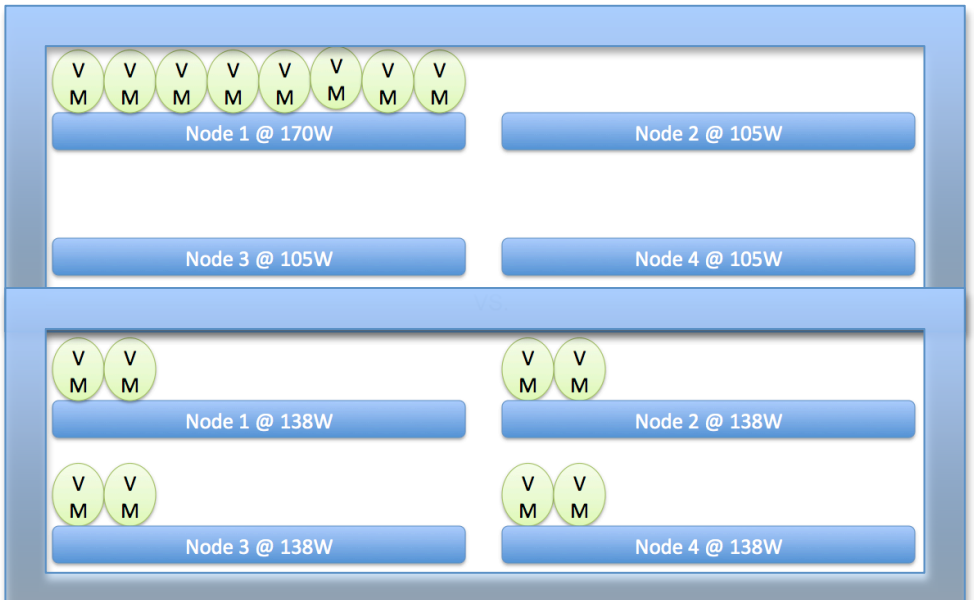
Framework



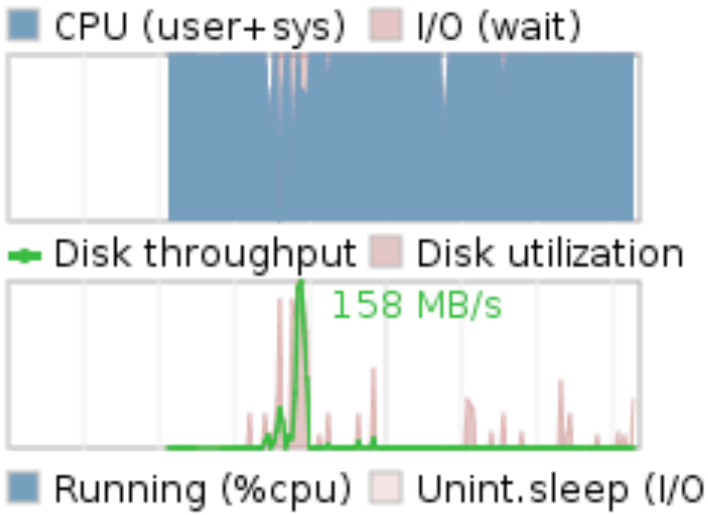
VM Efficiency in the Cloud

- Scheduling to minimize power consumption of the data center infrastructure.
- Minimizing Virtual Machine images to be lighter and faster.

485 Watts vs. 552 Watts



Booting Linux in 8 seconds



Open Science Grid

Simulating a Data Center

- Design a Simulator for simulating various cloud deployments in different types of data centers.
  - Model Hardware
  - Model Software
  - Model Workloads
- Hope to determine the most efficient type of cloud data center for both a given specific workload and a generalized workload.

Acknowledgements and Accomplishments

Special thanks to:

- Gregor von Laszewski
- Lizhe Wang
- Sonia Lopez Alarcron
- Pengcheng Shi
- Geoffrey Fox

Related Publications:

L. Wang, G. von Laszewski, A. Younge, X. He, M. Kunze, and J. Tao, Cloud Computing: A Perspective Study, in New Generation Computing, to appear in 2010.

L. Wang, G. von Laszewski, J. Dayal, X. He, A. Younge, and T. Furlani. "Towards Thermal Aware Workload Scheduling in a Data Center," in 10th International Symposium on Pervasive Systems, Algorithms and Networks (IS- PAN2009). Kao-Hsiung, Taiwan: Dec. 2009.

G. von Laszewski, L. Wang, A. Younge, and X. He, Power-Aware Scheduling of Virtual Machines in DVFS-Enabled Clusters, IEEE Cluster, 2009. New Orleans, LA USA: Sep, 2009.

G. von Laszewski, A. Younge, X. He, K. Mahinthakumar, and L. Wang, Experiment and Workflow Management Using Cyberaide Shell, in 4th International Workshop on Workflow Systems in e-Science with 9th IEEE International Symposium on Cluster Computing and the Grid. IEEE, May. 2009.


Open Science Grid

Real-Time Divisible Load Scheduling for Cluster Computing

Anwar Mamat

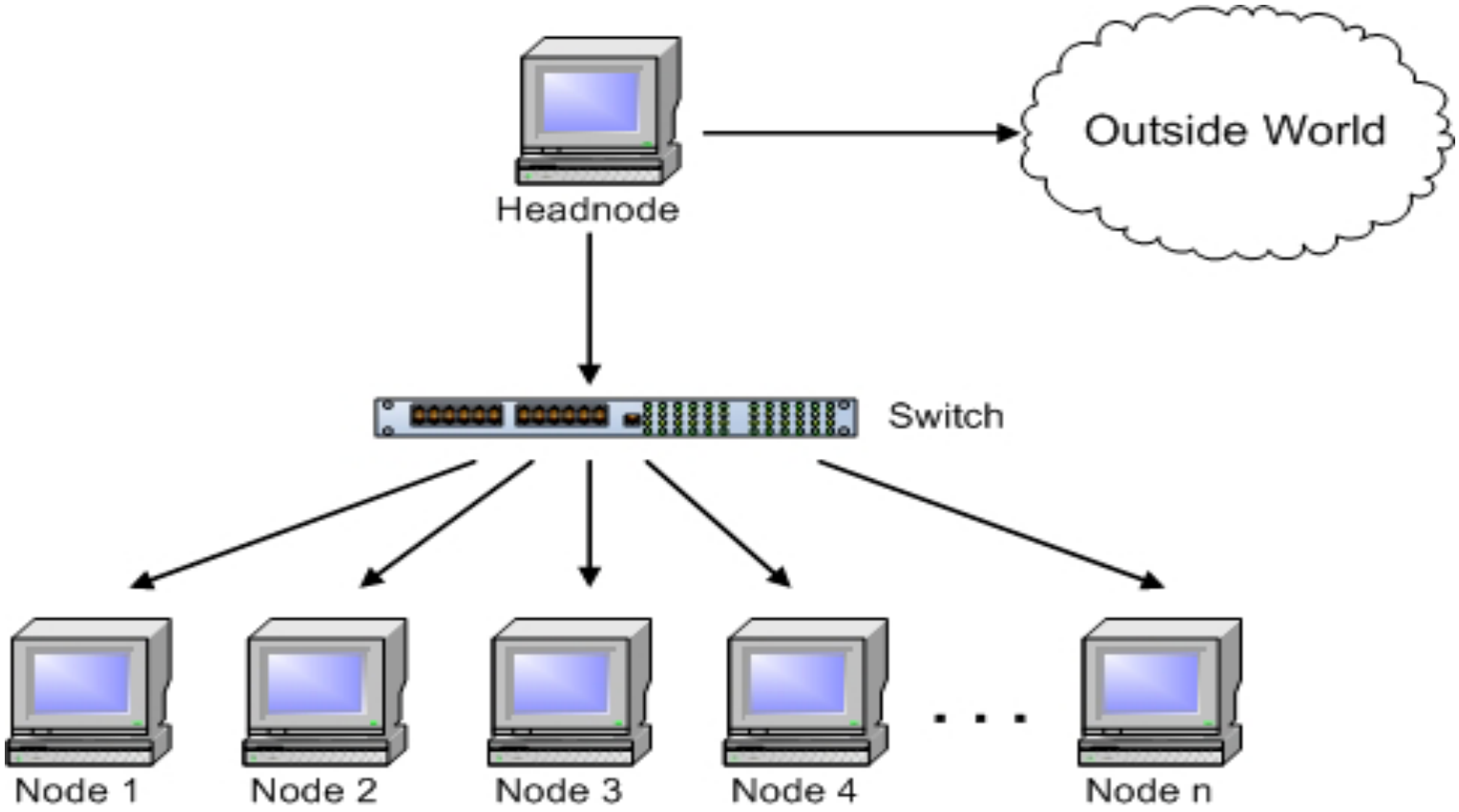
University of Nebraska

Motivation

- Providing QoS or real-time guarantees for arbitrarily divisible applications in a cluster
- Existing real-time cluster scheduling assumes task graph which is not appropriate for arbitrary divisible loads.
- Study the effects of the different design parameters

Open Science Grid

Real-time Cluster Computing

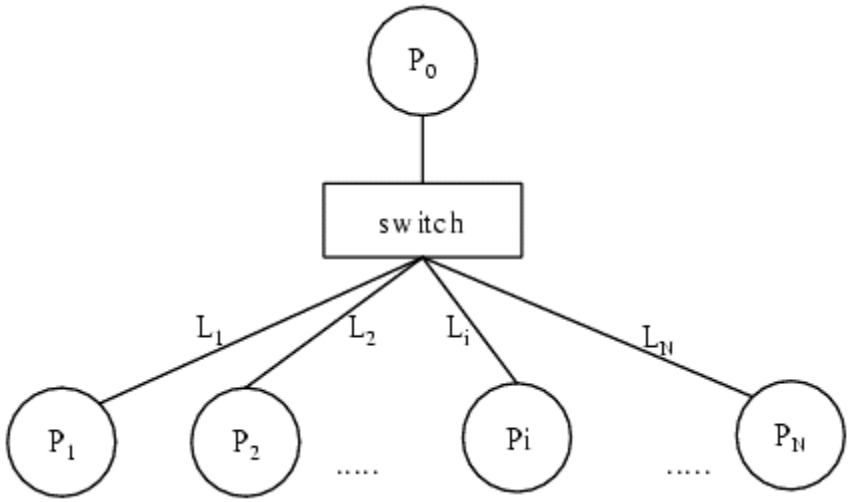


Open Science Grid

Task and System Model

- **Arbitrarily Divisible Task:** $T_i (A_i, \sigma_i, D_i)$
  - traditional real-time aperiodic task model: $T_i (A_i, C_i, D_i)$

- **System Model:**
  - C_{ms}, C_{ps}
  - $C_i = \varepsilon(\sigma_i, C_{ms}, C_{ps}, n)$



Open Science Grid

Algorithm

- Admission Controller and Dispatcher
- Real-time divisible load scheduling algorithm makes 3 important decision
  - Scheduling Policy
    - FIFO, MWF, EDF
  - Number of Processing Nodes
    - All , K, Min
  - Task Partition among the nodes
    - EPR, OPR

Open Science Grid

OPR

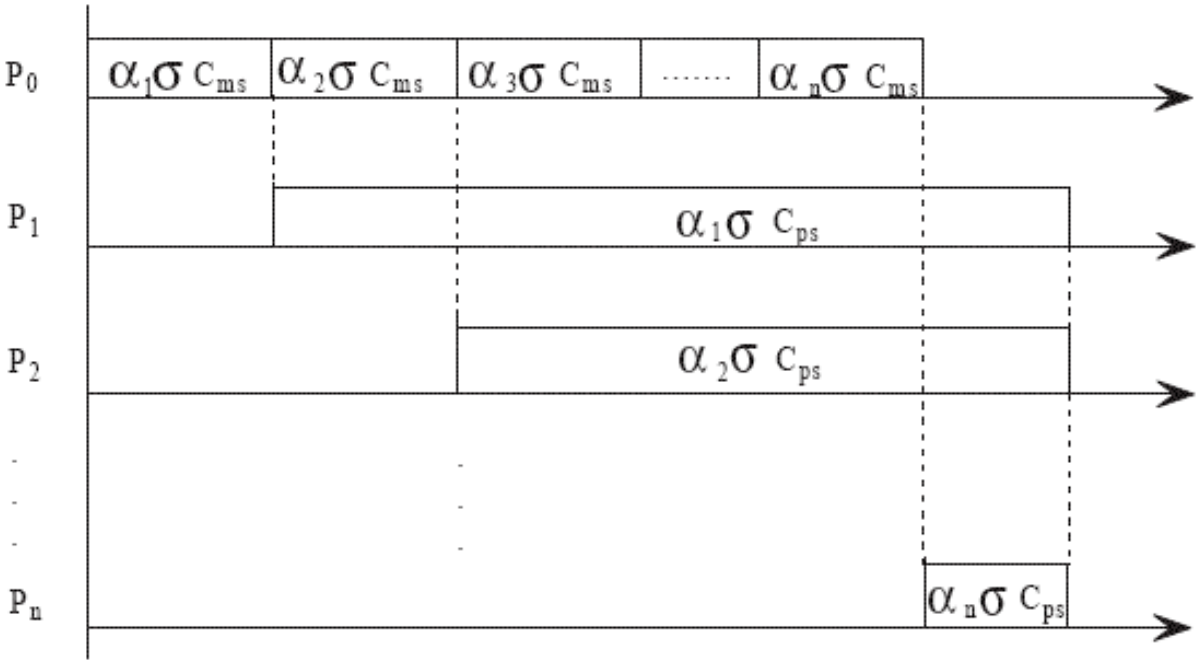


Figure 1: Time Diagram for OPR-Based Partitioning.

Open Science Grid

OPR cont.

$$\mathcal{E}(\sigma, n) = \frac{1-\beta}{1-\beta^n}\sigma(C_{ms} + C_{ps})$$

$$n^{min} = \lceil \frac{\ln \gamma}{\ln \beta} \rceil$$

$$\beta = \frac{\sigma C_{ps}}{\sigma C_{ms} + \sigma C_{ps}} = \frac{C_{ps}}{C_{ms} + C_{ps}}$$

$$\gamma = 1 - \frac{\sigma C_{ms}}{A+D-s}$$

Open Science Grid

Algorithms cont.

- Real-time divisible load scheduling with Advance Reservation
- Efficient real-time divisible scheduling
- Feedback control based real-time divisible load scheduling
- …

Open Science Grid

Conclusion

- Investigated
  - real-time divisible load scheduling in cluster environment
- Proposed
  - Several real-time divisible load scheduling algorithm
- Studied
  - effects of the different design parameters via **simulations**


Open Science Grid

Porting Bioinformatics to the OSG

Derek Weitzel

University of Nebraska

Open Science Grid

Derek Weitzel

- University of Nebraska – Lincoln
  - MS in Computer Engineering
- Went to ISSGC ’09 in Nice, France

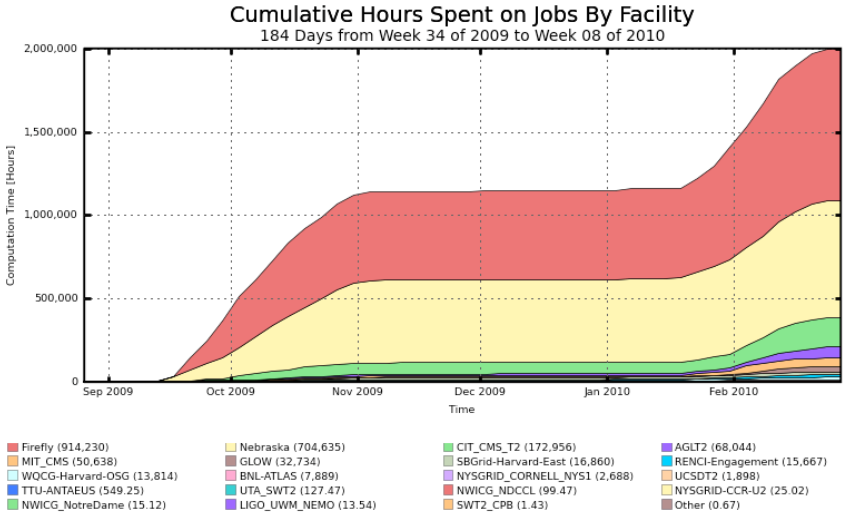
Open Science Grid

Use of OSG

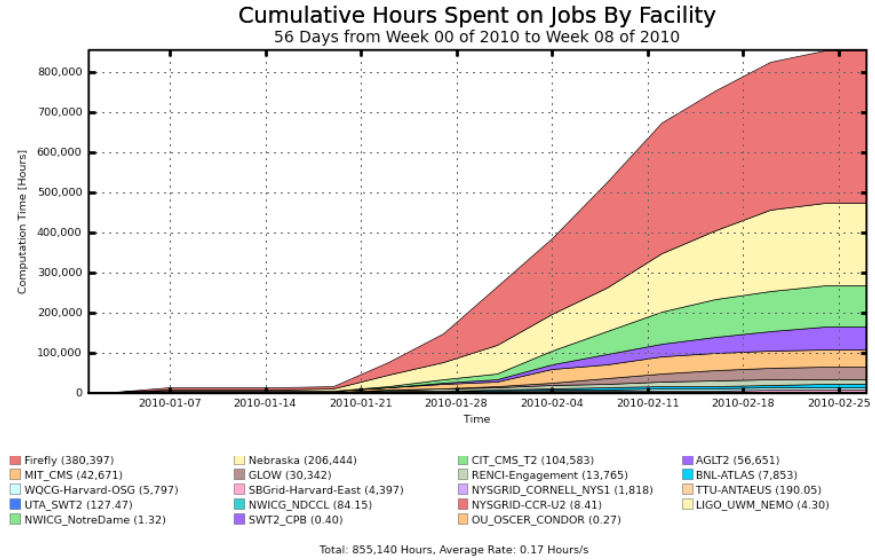
- Work with bioinformatics researchers to port applications
  - Rosetta & Autodock - OSGMM/GlideinWMS
  - DaliLite - GlideinWMS
- Class Projects
  - Effect of Queue length on throughput and X-Factor
- Masters Thesis
  - Performance Aware Grid Scheduling

Use of OSG

Primarily Nebraska resources

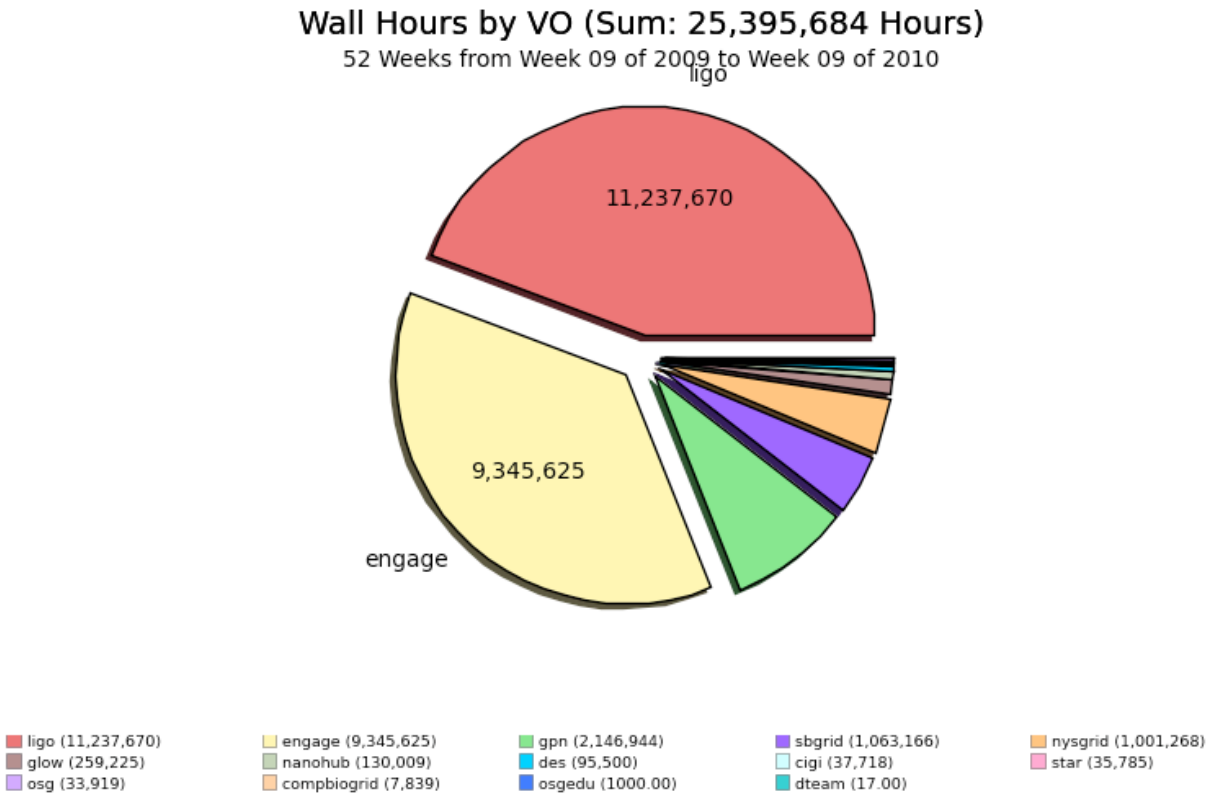


Recent diversified usage



Use of OSG

GPN (me) third largest non-HEP VO by computation hours




Open Science Grid

Experiences and Difficulties Implementing a Cluster in an Unprepared Environment

Cole Brand

University of Houston - Downtown

Session End

More Detailed Information

Real-Time Divisible Load Scheduling for Cluster Computing

Anwar Mamat

University of Nebraska

Open Science Grid

Scheduling Policies

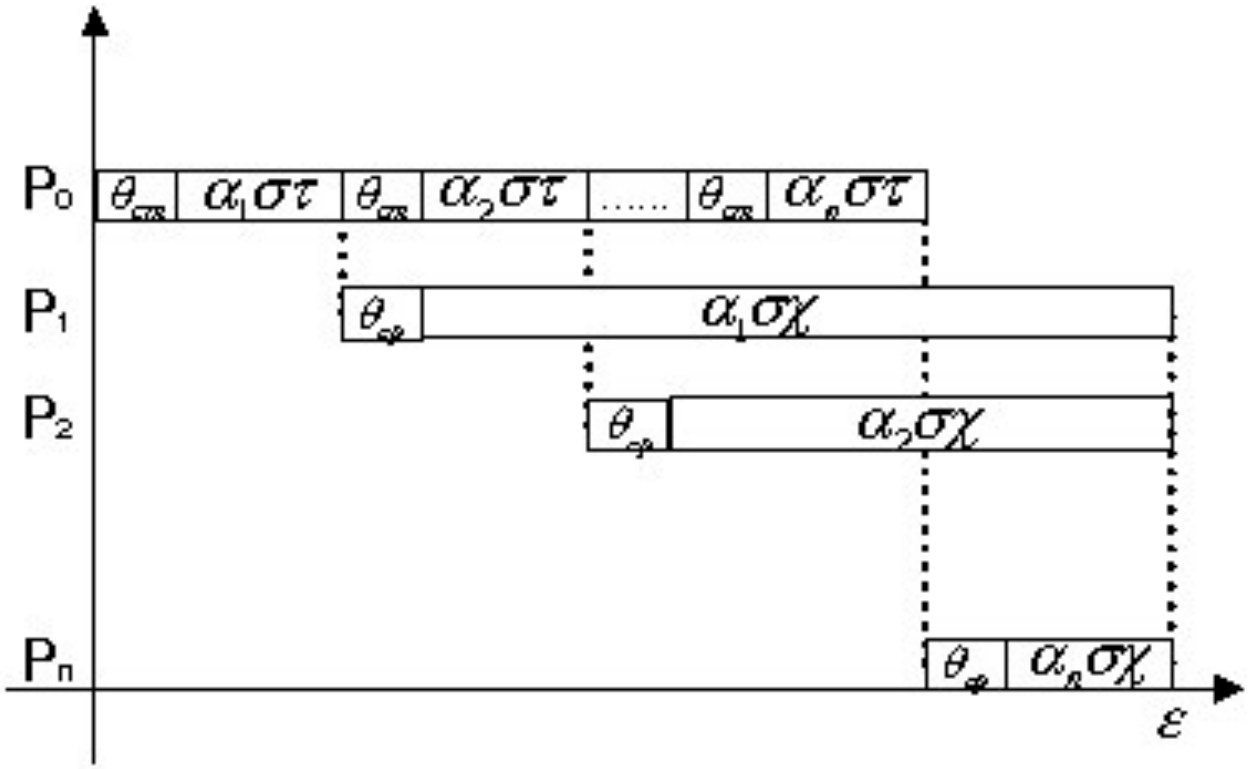
- FIFO
- MWF
  - MWF determines task execution order with the workload derivative metric, DCi.
  - Wi(n) workload (cost) of a task Ti when n processing nodes are assigned to it.

$$DC_i = W_i(n_i^{min} + 1) - W_i(n_i^{min})$$

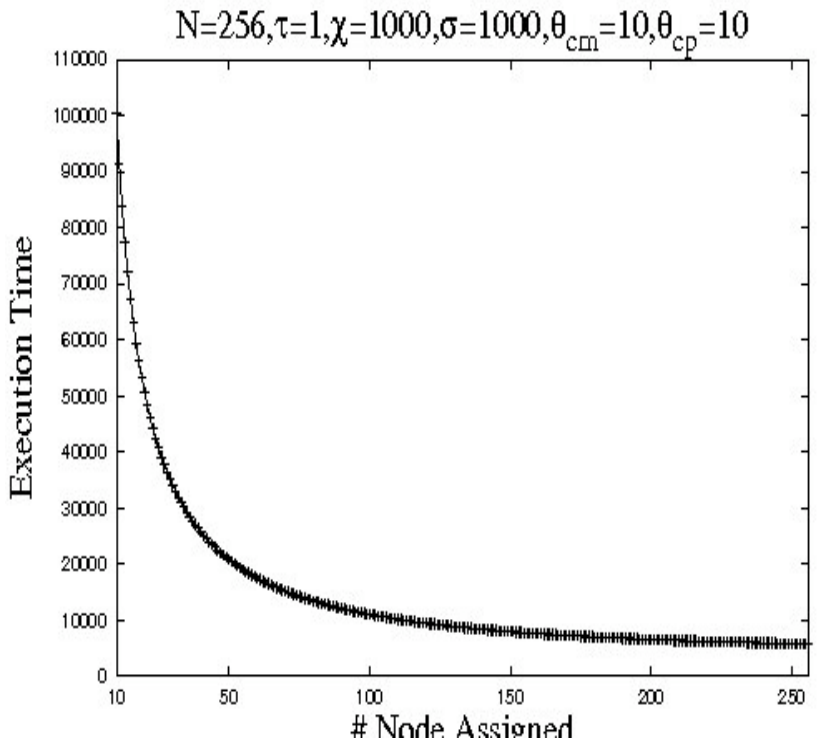
- ED $W_i(n) = n \times \mathcal{E}(\sigma_i, n)$

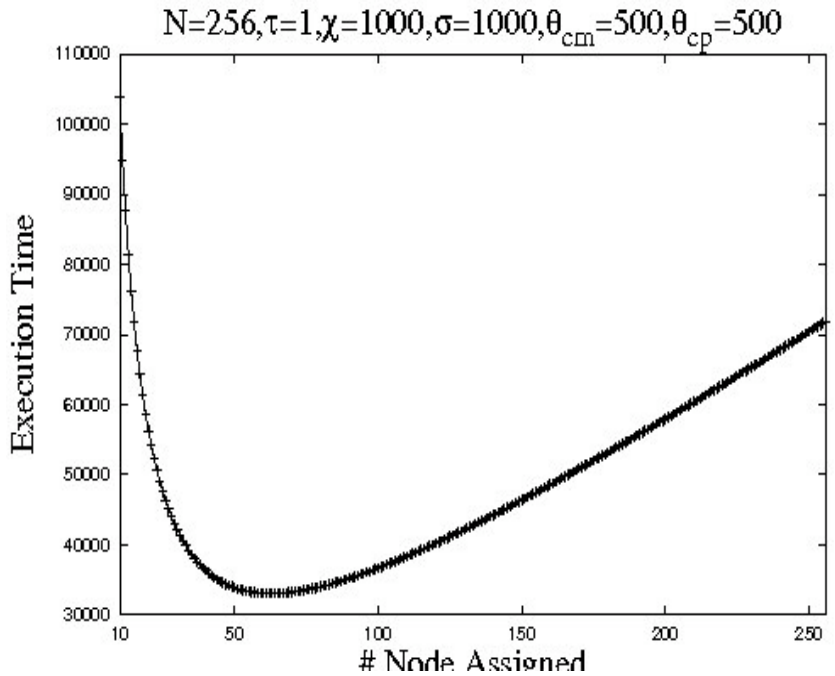
Open Science Grid

OPR cont.



OPR Based Partitioning with Setup Costs

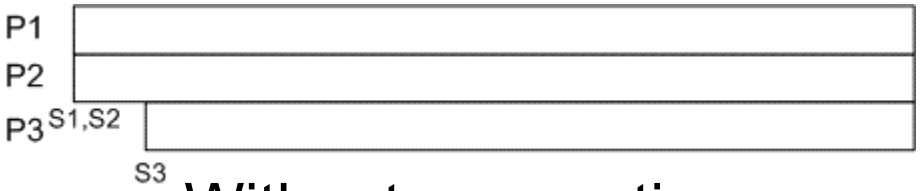

N=256,τ=1,χ=1000,σ=1000,θcm=10,θcp=10
Execution Time
Node Assigned
110000
100000
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
10
50
100
150
200
250


N=256,τ=1,χ=1000,σ=1000,θcm=500,θcp=500
Execution Time
Node Assigned
110000
100000
90000
80000
70000
60000
50000
40000
30000
10
50
100
150
200
250

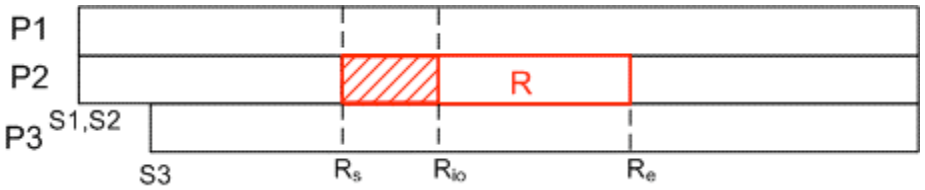
Open Science Grid

Advance Reservation Support

- **Nodes status:**



Without reservation

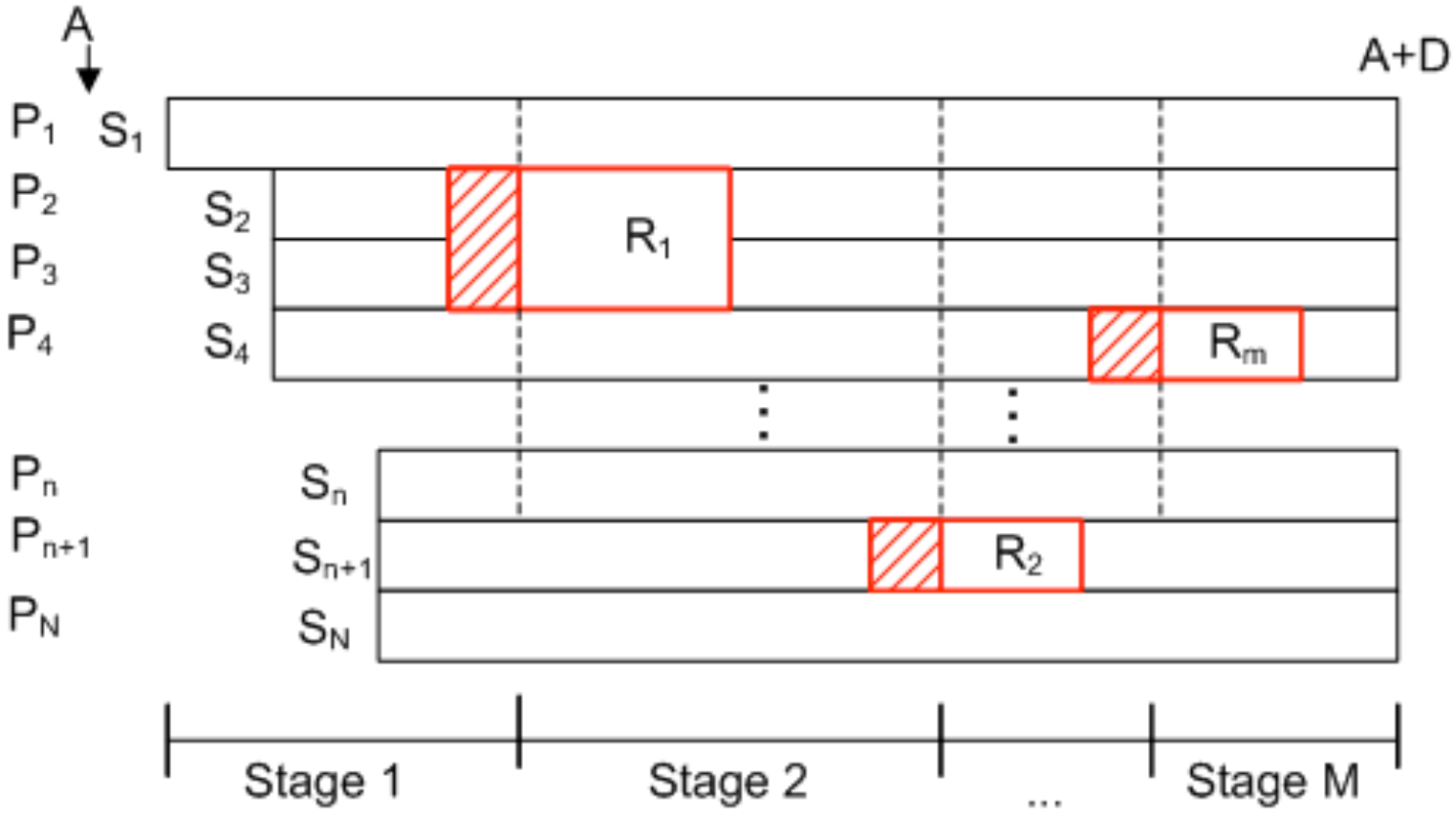


With reservation

Open Science Grid

Multi-Stage Task Partition

- **Divide nodes area into multiple stages:**



Efficient Real-time Scheduling Algorithm

- Relaxes the tight coupling between the admission controller and dispatcher.
- Linear time O(max(N, n)) complexity

Feedback Control Based Real-time Divisible Load Scheduling

- Handles the system variations dynamically
- Handles task execution time uncertainty
- Achieves high system utilization for soft real-time tasks

Experiences and Difficulties Implementing a Cluster in an Unprepared Environment

PRESENTER: COLE BRAND

ACADEMIC AFFILIATION:

UNIVERSITY OF HOUSTON – DOWNTOWN

DR. H. LIN AND DR. O. SIRISAENGTAKSIN

Open Science Grid

Abstract

- A short narrative of what it was like to install a cluster in an academic setting where there were no prior subject experts
- This presentation provided for those who were not able to attend the talks at the OSG All Hands meeting
- Contact details are provided at the end if you have further questions

Introduction

- I was initially associated with the project by attending a distributed computing / pthreads class at my university. The professor who was leading the class had been trying for years to get a campus compute cluster setup, and he recruited several students from this class to help. I offered to lend a friendly ear to help with problems that might come up. The team of 6 then tried to implement a full cluster by themselves, building everything from scratch.

Open Science Grid

A Beginning

- My team mate and I joined the project in the spring semester, rather virginal to the entire structure. We were aware that there were a number of computers available, and that we had a dedicated lab, and that we would be the only two working on the project. Aside from that information and our previous classroom experience, we had no foresight as to what lay in wait for us.

Open Science Grid

The “Lab”

- What we discovered initially was that the lab we would be using was the campus "Starcraft lab" - the place where sophmores and juniors would hang out and play video games (such as Starcraft) or card games (Magic: the Gathering). This lab was a haphazard assortment of computers, network equipment, simple Rubbermaid wire racks, a pair of beefier servers, and several desks and chairs. It was not laid out as an academic lab would hope to be laid out. After a short relayout of the lab, during which we found several gigabit switches that had been inadvertently lost, we had a room that allowed us to pursue the actual goal of getting a cluster setup and configured. We reorganized and consolidated the hardware of the roughly 60 commodity desktops of varying vintage into a gigabit backed network that eventually topped 80GFLOPs, which was fascinating to us, until we realized that the single server we were using as the head node was capable of an

Open Science Grid

Let's build a cluster

- We chose to go with the Rocks distribution for simple cluster building, and within two weeks of the beginning of the semester we had accomplished what four years of previous students had not been able to accomplish. Now we could move to the next round of what the professor wanted. Here were the primary goals:
  - Graphical user interface as opposed to a CLI. He was too worried about the learning curve of using the command line for the majority of future users.
  - Ease of user administration for using the cluster. How can we allow new users to work with the cluster?
  - Remotely accessible cluster. He wanted faculty and the like to be able to access any results remotely.

Did we meet the requirements?

- We were able to meet most of those goals using off-the-shelf components, such as Webmin, Usermin, and Apache with WebDAV. However, setting up a cluster is easy compared to the rest of what we had to go through.

Open Science Grid

Where's the IT guy?

- To begin with, we had no support from campus IT. The school is not a research heavy institution, having no graduate studies in the sciences, so this was not a priority for them. They actually wanted us completely off the campus network, to the point of assigning us an outside DSL. This meant we couldn't use the school's federated SSO, hence the #2 need previously to easily monitor who could access the system.

Open Science Grid

Chickens and eggs

- Additionally, and probably more importantly than the IT situation, we had no problems to run on the cluster. Granted it is a chicken-and-egg problem, but we should have a problem to run on the grid. As it is, we had a prior project from the lead professor to test the capabilities of the cluster but for which results were already known. We also were able to reimplement and speed up a project from a math professor, which allowed her to gain results, but the project wasn't directly grid-designed in the first place.

Open Science Grid

The worst problem…

- Probably what I would say was the worst problem we had, was the lack of a local domain expert. We had nobody on campus who could tell us how to setup a cluster, or what the purpose of a cluster was, or how to design one, or what to do with one. Everything that we learned about clusters was from reading online, self instruction, and collaboration with online user groups. This was nice, but my partner and I could tell something was missing. Our project had no raison d'être, and upon discussing this with our advisor, he issued a new project goal, namely connecting our grid to the OSG, which he had some experience with. Ah! A lead. So we began by checking the OSG website, and we were still confused. How did one go about joining the OSG? So I'm going to skip the mundane details about emails and the like, but long story short, I got hooked up with the OSG and we started talking about what it takes to get the school involved with the OSG project.

Open Science Grid

First Contact of an OSG kind

- Following getting in touch with OSG, I was later invited to ISSGC'09, and my team-mate was invited to TeraGrid. We learned a great deal more about the purpose and focus of distributed, cluster and grid computing, and we both brought back specific insights. My insight was that all the things we were hoping to accomplish were usually only done by graduate students in research universities with local domain experts and support staff (project IT, undergrads for grunt-work, funding for larger facilities). So I felt pretty good that we had gotten done what we had, but I knew that we had barely done the equivalent of stacking a small pile of pebbles in the grand scheme of things that could be done. "Standing on the shoulders of giants"

Open Science Grid

Summation

- So in the end, the lack of a local domain expert was the biggest problem with our cluster experience. The facility is still able to run calculations, it can still be used for training purposes or for faculty who want to use it, but it's pretty much just a toy. If OSG is looking for a way to reach out to institutions, I can't really suggest much more than to make it easy for organizations like ours to reach out to them for guidance. That's definitely something I've learned that the OSG group is looking to do. But additionally, it's not just necessary to link up with the CS groups, but rather, the CS groups that want to participate need some help in knowing what to

Open Science Grid

Followup example

- Here's my example for that. Three of our non-CS sciences faculty had experience with and desire to use local resources for distributed computing. But until we had direction from the Rocks team (BLAST and other tools being deployed in rolls, our asking the sciences group what those programs were, they getting excited we were asking) we didn't even know the questions to ask our own faculty. I think that having a local purpose for distributed computing, ours not being a research facility, would have been a good catalyst to spurring the self feeding cycle on. This might be something that OSG can assist in: setting up a program for ancillary sites that just want to train students in distributed computing in preparation for graduate studies.

Finale, Contact

- That's the nutshell of my experiences with setting up a cluster computing effort at my university and my connection with OSG. It's brief, but hopefully it carries some insight.

- If you have any further questions, feel free to contact me at j.cole.brand+OSG@gmail.com.

Open Science Grid

Materials to work with

- Equipment
  - 60+ Dell desktop
    - 18 Pent Dual-Core / 1GB
    - 18 Pent D / 512MB
    - 34 Pent 4 / 256MB
  - 4x 16pt 100MB switches
  - 3x 24pt Gb switches
- Lab
  - The “Starcraft” lab
  - Corner office with poor ventilation
  - Copious amounts of unknown equipment
- Previous team had tried
  - To setup 3 smaller clusters
  - Poor network architecture
  - To write their own management routines
  - No strong Linux members
- School IT department
  - Doesn’t care about this project

School Staff | Intended use

- CS professors
  - Only 3 with strong cluster background
  - No clear focus or purpose
- Math professors
  - Two with strong distributed computing background
- Nat Sci professors
  - Two with previous dist. computing background

- Intended software to run
  - None.
- Intended standardized Linux distribution
  - None.
- Intended lifetime of cluster
  - Indeterminate
- Anticipated professional member organizations

Intended Purpose | Setup Decisions

- Target Audience
  - Juniors with no exp
  - Senior Projects
  - Math faculty
  - CS faculty
  - Nat Sci faculty
- Preferred method of use
  - Everyone wants web interfaces
  - NAS based */home* dir
- How we set things up
  - Single large network
  - Rocks 5.1
  - Refitted P4s to 512MB
- UI
  - SSH
  - Webmin / Usermin
- Languages
  - Java
  - C/C++
  - R